

What next, now that the Cloud has failed....?



Gary Newe, 2011-28-04

The outage at Amazon 'surprisingly' took a lot of large profile sites with it. 'Surprisingly' in the sense that it is surprising that these sites didn't have a back-up plan.

Working in IT adds a third certainty to taxes and death: things will fail! That is why we take backups. That is why we plan for failures, and that is why we never, never rely on a single point of failure. This is why the business continuity (BC) [market is expected to be worth more than \\$39 billion by 2015](#).

So what went wrong? Well, that is for Amazon to answer. It seems that there was a major failure that took down more than it should have.

But I don't really blame Amazon. Things fail and we all know better now than to rely on a single company or a single solution to run our mission critical or business critical applications.

A lot of the sites that were affected by the Amazon outage, like Reddit (owned by Conde Nast), and Heroku (owned by Salesforce.com), need to re-examine the way they approach BC. These guys are not lacking in IT expertise – they are not small start-ups.

Other than the term itself, I am not against cloud. I think that the idea of cloud is amazing: computing on demand and shared services are very attractive concepts from a service and cost point of view.

But if all you rely on is a single service provider, then when it fails it's time to ask the question "[Pop quiz, hot shot, what do you do?](#)"

And, in a roundabout way, this brings me to the point.

What I think we should look at to solve this issue and still be able to take advantage of cloud is touched on by some discussion on the web about what people are terming a "Super Cloud" or "The Cloud of Clouds".

This entails using more than one cloud vendor, move your apps between vendors and use resource from different vendors based on usage cost, time of day etc. We are not there yet but if you are clever about your own infrastructure you can get close to that today.

You need to use your application delivery controller (ADC) for what it was designed for, to deliver your applications: to bridge the gap between your applications and your users.

Imagine, if you will, that you could run your applications in your own data centre then, when you get a surge in users, burst your capacity to your cloud vendor, whether it is Amazon, RackSpace or any other cloud provider.

What if you could detect failures in your infrastructure or your provider's infrastructure, and re-provision to another cloud provider? The ultimate aim should be to keep your applications available. So why not put your ADC to use to detect failures in your applications wherever they are provisioned, respond to these failures and mediate to fix these failures?

I suggest you have a read of this series on dynamic infrastructure, <http://devcentral.f5.com/weblogs/npearce/Default.aspx> where Nathan goes into some detail on this topic.

I leave you with this, which is a sobering lesson in how not to do High Availability, particularly for critical services: <https://forums.aws.amazon.com/thread.jspa?threadID=65649&tstart=0>

If you fail to prepare, prepare to fail.

Technorati Tags: [f5](#), [F5 Networks](#), [Security](#), [Amazon](#), [Cloud](#), [EC2](#), [Amazon Outage](#), [Amazon Failure](#), [BC](#), [DR](#), [Disaster Recovery](#), [Speed](#), [#fail](#), [fail](#)

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

©2016 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com. Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113